|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Haofei Yan | hy222ap | hy222ap@student.lnu.se | Huan Rong | hr222dx |  |  | https://github.com/bigmachuan/1DV607.git |  |  |  |  |
| Kristoffer Karlsson | kk222hk | kristofferkarlsson91@gmail.com |  |  |  |  | https://github.com/kk222hk/1dv607-kk222hk-Portfolio/tree/master/Workshop%202 |  |  |  |  |

Peer Review Instructions Workshop 2 Design

The peer review is done for grade 2 only.

You perform the review in the same group that you handed had when your materials for grade 2. Group contact person, will get an email with groups to review and then forwards this to the group members. After that you organize the review.

You are to create and hand in a review of 2 other groups. The goal is to give good feedback to these groups and for you to get different points of view when reviewing your own design/implementation.

Read the theory chapters in the book and other sources. You will need to **motivate your opinions with proper references**. You can not just state “we think that X is good/wrong”, or “Tobias said X was good/wrong”, rather use something like: “Larman states [1, p123] that in order to model X you should think about…this is also supported by Fowler [2]”

Your peer review **should have a References section**. In this section references should be numbered in the following form:

Nr, Author, Title, Year/Date, ISBN/link

For example:

1. Larman C., Applying UML and Patterns 3rd Ed, 2005, ISBN: 0131489062
2. Fowler M., Presentation Model, 2015-08-26, http://martinfowler.com/eaaDev/PresentationModel.html

Your references should focus on original references (for example if you find something on wikipedia, read the original source), blogs etc. can be ok if they are based on a serious source/well recognized name/organisation.

Your review should be in the same language that the materials you receive are made in.

If you are doing the review in a group, all members should do the steps together and discuss their point of view for each question.

Add the reviews to your own portfolio and prepare public links.

Send your review links to the respective group contact persons when you are done.

When you  have received your reviews fill in the form on the course home page.

**Instruction**

Look at the models, implementation and any accompanying documentation. Try to have an open mind and focus on trying to understand the materials as it is presented.

Test the runnable version of the application in a realistic way. Note any problems/bugs.

Try to compile/use the source code using the instructions provided. Can you get it up and running? Is anything problematic? Are there steps missing or assumptions made?

Does the implementation and diagrams conform (do they show the same thing)? Are there any missing relations? Relations in the wrong direction? Wrong relations? Correct UML notation?

Is the Architecture ok?

* Is there a model view separation?
* Is the model coupled to the user interface?
* Is the model specialized for a certain kind of IU (for example returning formated strings to be printed)
* Are there domain rules in the UI?

Is the requirement of a unique member id correctly done?

What is the quality of the implementation/source code?

* Code Standards
* Naming
* Duplication
* Dead Code
* ...

What is the quality of the design? Is it Object Oriented?

* Objects are connected using associations and not with keys/ids.
* Is GRASP used correctly?
* Classes have high cohesion and are not too large or have too much responsibility.
* Classes have low coupling and are not too connected to other entities.
* Avoid the use of static variables or operations as well as global variables.
* Avoid hidden dependencies.
* Information should be encapsulated.
* Inspired from the Domain Model.
* Primitive data types that should really be classes (painted types)
* ...

As a developer would the diagrams help you and why/why not?

What are the strong points of the design/implementation, what do you think is really good and why?

What are the weaknesses of the design/implementation, what do you think should be changed and why?

Do you think the design/implementation has passed the grade 2 criteria?